**BÁO CÁO THỰC HÀNH IoT**

Tên nhóm: Nhóm 1

Tên thành viên:

* Sử Thành Công
* Hồ Thị Thanh Ngân
* Trần Cao Thục Uyên
* Đặng Văn Thắng
* Trần Đoàn Ngọc Duy
* Mai Thị Phúc Trâm

**Mục lục**

[Bài thực hành 1: Khái niệm về IoT 1](#_Toc122683779)

[1.1. Tìm hiểu thiết bị và cài đặt phần mềm lập trình 1](#_Toc122683780)

[1.2. Điều khiển On/Off led trên ESP và hiển thị qua serial (1) 1](#_Toc122683781)

[**1.2.1.** **Linh kiện** 1](#_Toc122683782)

[**1.2.2.** **Thực hành** 1](#_Toc122683783)

[Bài thực hành 2: Thiết lập ban đầu cho dự án IOT 2](#_Toc122683784)

[2.1. Hiển thị nội dung trên LCD i2c và hiển thị qua serial 2](#_Toc122683785)

[**2.1.1.** **Linh kiện** 2](#_Toc122683786)

[**2.1.2.** **Thực hành** 2](#_Toc122683787)

[2.2. Kết nối biến trở đo adc và hiển thị qua LCD i2c và serial (2) 4](#_Toc122683788)

[**2.2.1.** **Linh kiện** 4](#_Toc122683789)

[**2.2.2.** **Thực hành** 4](#_Toc122683790)

[2.3. Kết nối cảm biến (DHT11/DHT22) với ESP8266 và hiển thị nhiệt độ độ ẩm qua LCD i2c và serial (3) 6](#_Toc122683791)

[**2.3.1.** **Linh kiện** 6](#_Toc122683792)

[**2.3.2.** **Thực hành** 6](#_Toc122683793)

[2.4. Điều khiển relay và hiển thị qua LCD i2c và serial 8](#_Toc122683794)

[**2.4.1.** **Linh kiện** 9](#_Toc122683795)

[**2.4.2.** **Thực hành** 9](#_Toc122683796)

[Bài thực hành 3: Giám sát dữ liệu đám mây 10](#_Toc122683797)

[3.1. Lấy dữ liệu trực tuyến (dữ liệu bất kỳ) hiển thị qua LCD i2c 10](#_Toc122683798)

[**3.1.1.** **Linh kiện** 10](#_Toc122683799)

[**3.1.2.** **Thực hành** 10](#_Toc122683800)

[3.2. Kết nối cảm biến (DHT11/DHT22) với ESP8266 và hiển thị qua LCD i2c và lưu trữ dữ liệu trên Thingspeak (4) 14](#_Toc122683801)

[**3.2.1.** **Linh kiện** 14](#_Toc122683802)

[**3.2.2.** **Thực hành** 15](#_Toc122683803)

[Bài thực hành 4: Tương tác với các dịch vụ web và máy với máy 18](#_Toc122683804)

[4.1. & 4.2. Giám sát dữ liệu cảm biến từ bảng điều khiển đám mây(lấy kết quả từ 3.2.) 18](#_Toc122683805)

[**4.1.1. & 4.2.1.** **Linh kiện** 18](#_Toc122683806)

[**4.1.2. & 4.2.2.** **Thực hành** 18](#_Toc122683807)

[Tài liệu tham khảo 22](#_Toc122683808)

# Bài thực hành 1: Khái niệm về IoT

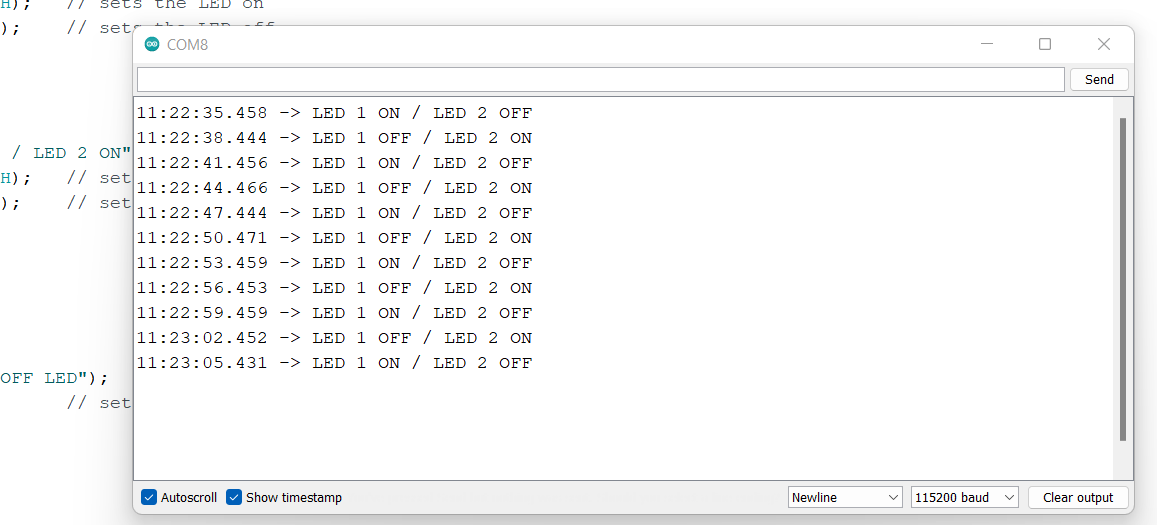
* 1. Tìm hiểu thiết bị và cài đặt phần mềm lập trình
  2. Điều khiển On/Off led trên ESP và hiển thị qua serial (1)

1. **Linh kiện**

* Esp8266
* Dây usb

1. **Thực hành**

* Màn hình Serial Monitor:



* Trình bày code:

int ledPin1 = 16;                 // LED kết nối với chân kỹ thuật số 16

int ledPin2 = 2;                 // LED kết nối với chân kỹ thuật số 2

void TURN\_ON\_LED1(void)

{

  Serial.println("LED 1 ON / LED 2 OFF"); // hiển thị trạng thái trên Serial

  digitalWrite(ledPin1, HIGH);   // bật đèn LED 1

  digitalWrite(ledPin2, LOW);    // tắt đèn LED 2

} // high : bật low : tắt

void TURN\_ON\_LED2(void)

{

  Serial.println("LED 1 OFF / LED 2 ON"); // hiển thị trạng thái trên Serial

  digitalWrite(ledPin2, HIGH);   // bật đèn LED 2

  digitalWrite(ledPin1, LOW);    // tắt đèn LED 1

}

void setup()

{

  Serial.begin(115200);

  Serial.println("BTH1: ON/OFF LED");// hiển thị trên màn hình LCD

  pinMode(ledPin1, OUTPUT);      // sets the digital pin as output

  pinMode(ledPin2, OUTPUT);

}

void loop()

{

  TURN\_ON\_LED1(); // đồng thời bật đèn LED 1 và tắt đèn LED 2 và ngược lại

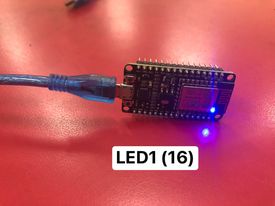
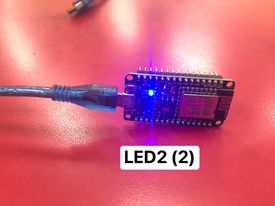
  delay(3000);     // chờ 3s

  TURN\_ON\_LED2();

  delay(3000);                  // chờ 3s

}

* Kết quả:



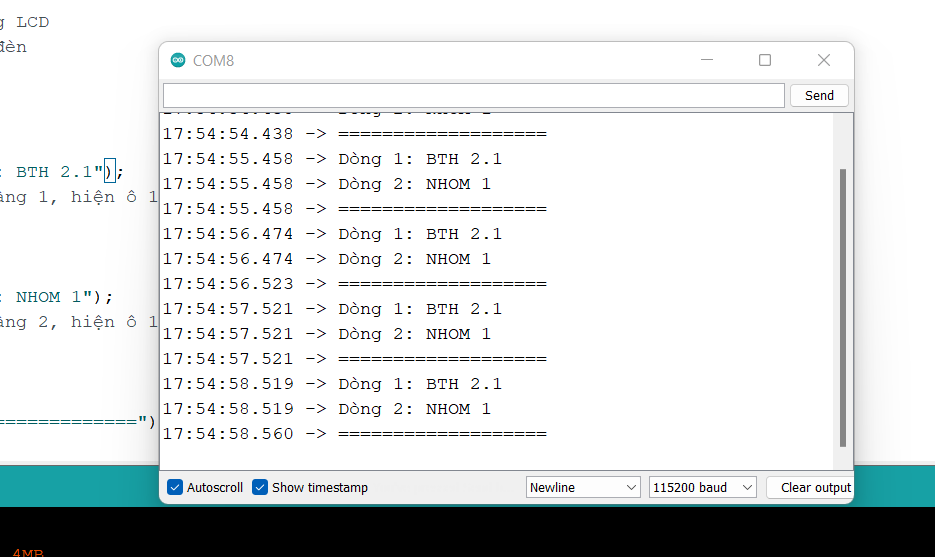
# Bài thực hành 2: Thiết lập ban đầu cho dự án IOT

1. Hiển thị nội dung trên LCD i2c và hiển thị qua serial
2. **Linh kiện**

* Esp8266
* Dây usb
* Màn hình LCD I2C 1602
* 4 dây cái-cái

1. **Thực hành**

* Màn hình Serial Monitor:



* Trình bày code:

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,19,20);

void setup() {

  //  Serial port 115200

  Serial.begin(115200);

  lcd.init(); //Khởi động LCD

  lcd.backlight(); //Mở đèn

}

void loop() {

  //Con trở ở vị trí 0 hàng 1, hiện ô 1

  lcd.setCursor(0,0);

  lcd.print("BTH 2.1");

  //Con trở ở vị trí 0 hàng 2, hiện ô 1

  lcd.setCursor(0,1);

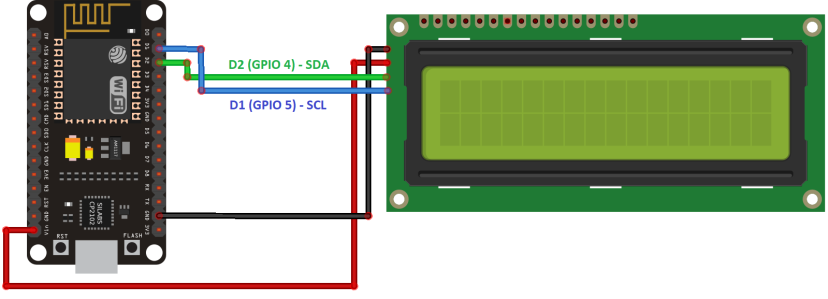
  lcd.print("NHOM 1");

  Serial.println("========================");

  delay(1000);

}

* Kết quả:





1. Kết nối biến trở đo adc và hiển thị qua LCD i2c và serial (2)
2. **Linh kiện**

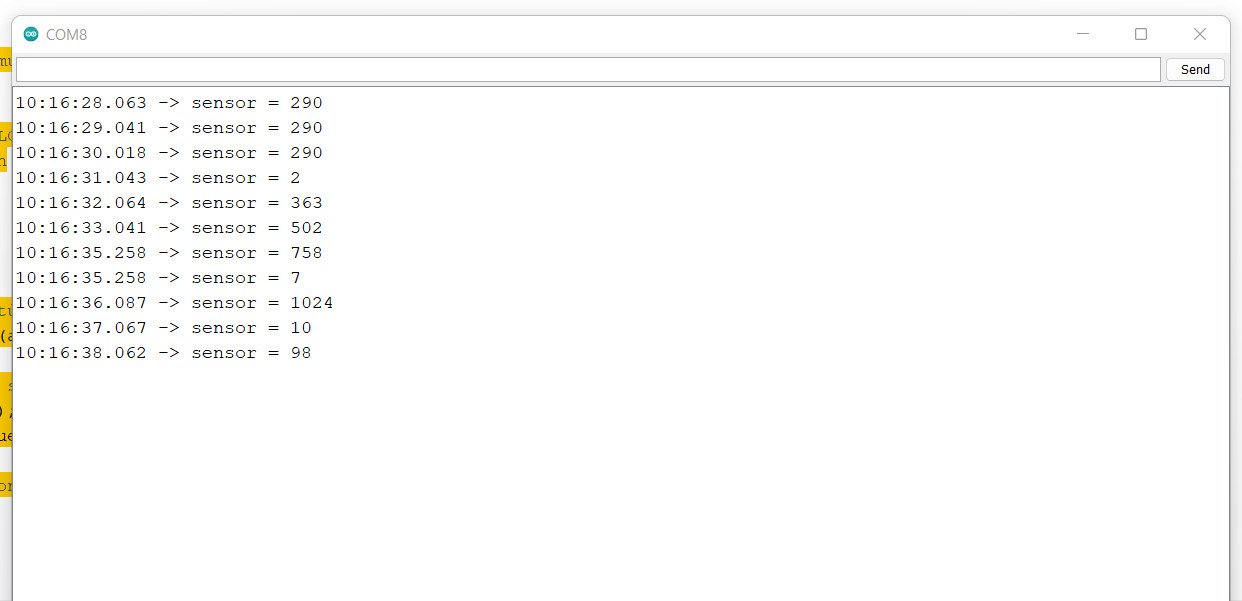
* Esp8266
* Dây usb
* Màn hình LCD I2C 1602
* Biến trở 20k



* 7 dây cái-cái

1. **Thực hành**

* Màn hình Serial Monitor:



* Trình bày code:

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,19,20);

const int analogInPin = A0;  // ESP8266 Analog Pin ADC0 = A0

int sensorValue = 0;

void setup() {

  //  serial port 115200

  Serial.begin(115200);

  lcd.init(); //Khởi động LCD

  lcd.backlight(); //Mở đèn

  lcd.setCursor(0,0); //Con trỏ ở vị trí 0, hiện ô 1

  lcd.print("Bien tro:");

}

void loop() {

  // Đọc giá trị Biến trở từ chân Analog A0

  sensorValue = analogRead(analogInPin);

  // Hiển thị kết quả trên serial

  Serial.print("sensor = ");

  Serial.println(sensorValue);

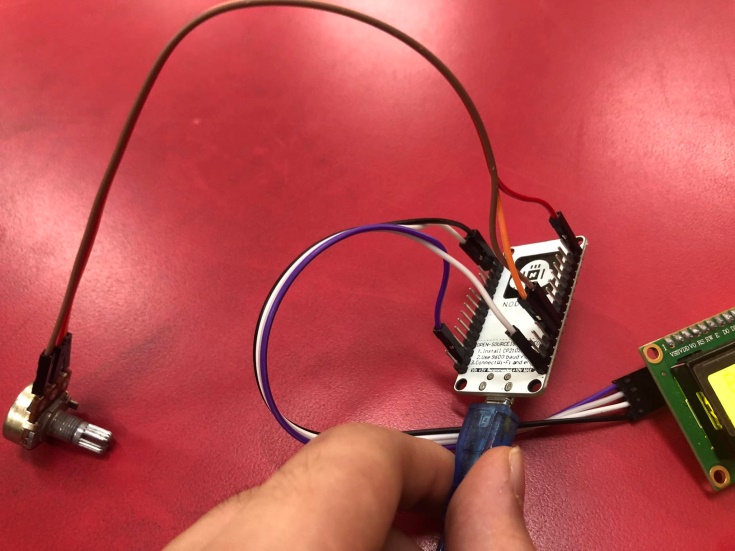
  lcd.setCursor(10,0); //Con trỏ ở vị trí 10, hiện ô 11

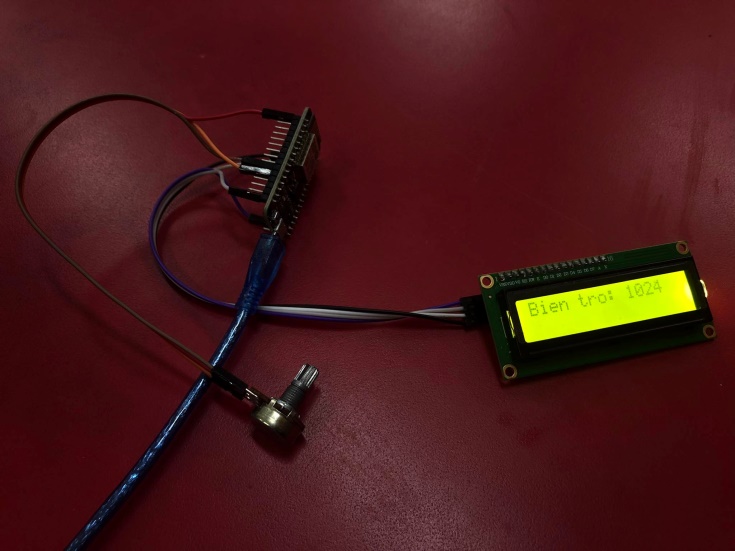
  lcd.print(sensorValue);

  delay(1000);

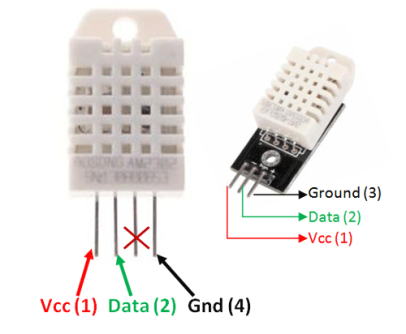
}

* Kết quả:



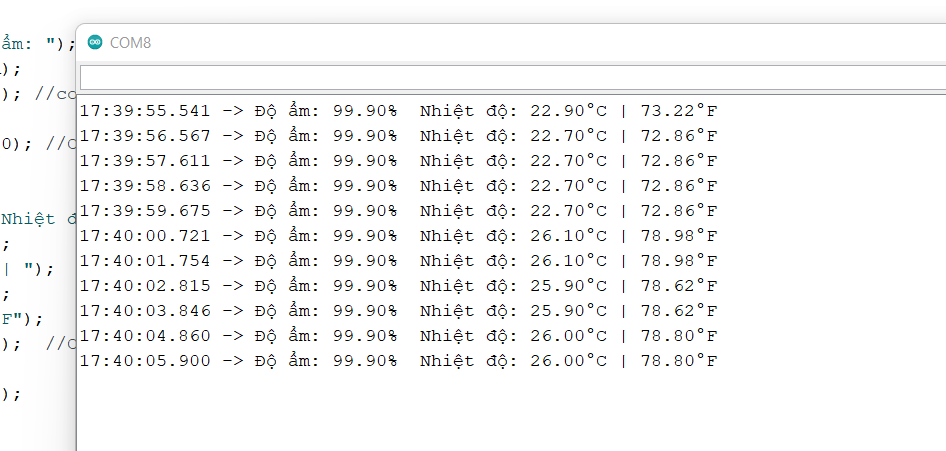


1. Kết nối cảm biến (DHT11/DHT22) với ESP8266 và hiển thị nhiệt độ độ ẩm qua LCD i2c và serial (3)
2. **Linh kiện**

* Esp8266
* Dây usb
* Màn hình LCD I2C 1602
* 7 dây cái-cái
* DHT22 

1. **Thực hành**

* Màn hình Serial Monitor:



* Trình bày code:

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,19,20);

#include <DHT.h>

const int DHTPIN = D5;// Chân Out của cảm biến nối chân số 5 ESP8266

const int DHTTYPE = DHT22;   // Khai báo kiểu cảm biến là DHT21

DHT dht(DHTPIN, DHTTYPE); //Khai báo thư viện chân cảm biến và kiểu cảm biến

void setup()

{

  Serial.begin(115200);

  dht.begin(); //Khởi động cảm biến

  lcd.init(); //Khởi động LCD

  lcd.backlight(); //Mở đèn

  lcd.setCursor(0,0); // con trỏ vị trí số 0 hàng 1, hiện ô số 1

  lcd.print("DO AM:");// hiển thị "DO AM:" trên LCD

}

void loop()

{

  float doam = dht.readHumidity(); //Đọc độ ẩm

  float doC = dht.readTemperature(); //Đọc nhiệt độ C

  float doF = dht.readTemperature(true); //Đọc nhiệt độ F

  // Kiểm tra cảm biến có hoạt động hay không

  if (isnan(doam) || isnan(doC) || isnan(doF))

  {

    Serial.println("Không có giá trị trả về từ cảm biến DHT");

    return;

  }

  Serial.print("Độ ẩm: ");

  Serial.print(doam);

  lcd.setCursor(7,0); //con trỏ vị trí số 7 hàng 1, hiện ô số 8

  lcd.print(doam);

  lcd.setCursor(12,0); //Con trỏ ở vị trí 12, hiện ô 13

  lcd.print("%");

  Serial.print("%  Nhiệt độ: ");

  Serial.print(doC);

  Serial.print("°C | ");

  Serial.print(doF);

  Serial.println("°F");

  lcd.setCursor(0,1);  //Con trỏ ở vị trí 0 hàng 2, hiện ô 1

  lcd.print(doC);

  lcd.setCursor(5,1); // Con trỏ ở vị trí 5 hàng 2, hiện ô 6

  lcd.print("'C-");

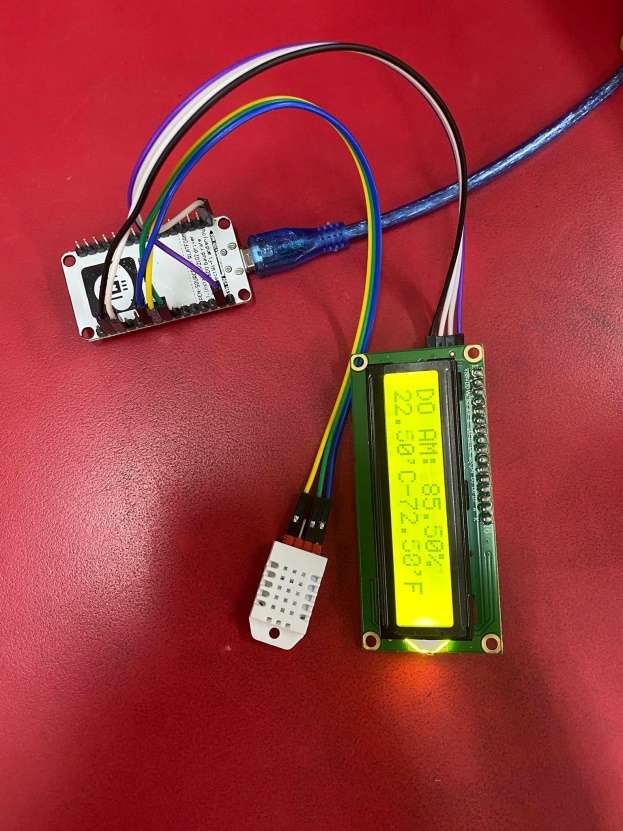
  lcd.print(doF);

  lcd.print("'F");

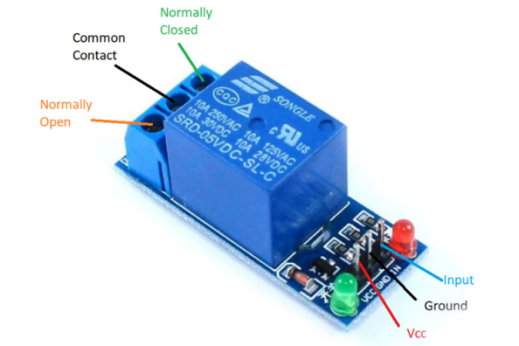
  delay(1000);

}

* Kết quả:

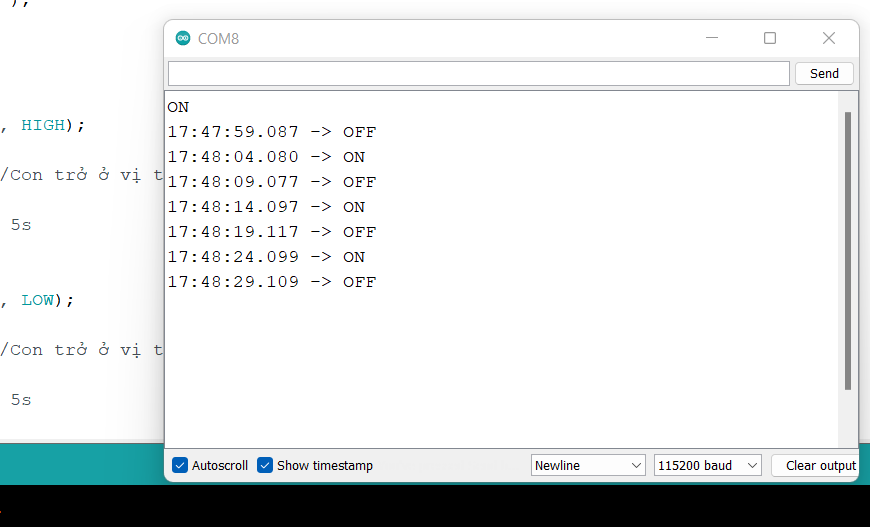


1. Điều khiển relay và hiển thị qua LCD i2c và serial
2. **Linh kiện**

* Esp8266
* Dây usb
* Màn hình LCD I2C 1602
* Relay 5v
* 7 dây cái cái

1. **Thực hành**

* Màn hình Serial Monitor:



* Trình bày code:

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,19,20);

#define RELAY\_PIN D5 // Relay nối với Pin D5

void setup() {

  Serial.begin(115200);

  pinMode(RELAY\_PIN, OUTPUT); // sets the digital pin as output

  lcd.init(); //Khởi động LCD

  lcd.backlight(); //Mở đèn

  lcd.setCursor(0,0);  //Con trở ở vị trí 0 hàng 1, hiện ô 1

  lcd.print("Trang thai:");

}

void loop() {

  /// bật

  digitalWrite(RELAY\_PIN, HIGH);

  Serial.print("ON");

  lcd.setCursor(12,0); //Con trở ở vị trí 12, hiện ô 13

  lcd.print("ON ");

  delay(5000);  /// Nghỉ 5s

  /// Tắt

  digitalWrite(RELAY\_PIN, LOW);

  Serial.print("OFF");

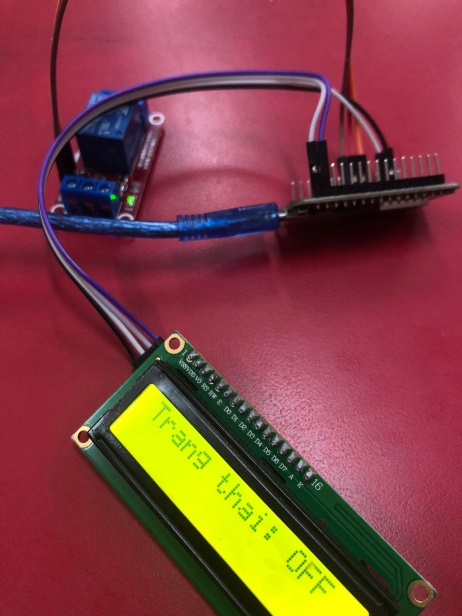
  lcd.setCursor(12,0); //Con trở ở vị trí 12, hiện ô 13

  lcd.print("OFF");

  delay(5000);  /// Nghỉ 5s

}

* Kết quả:



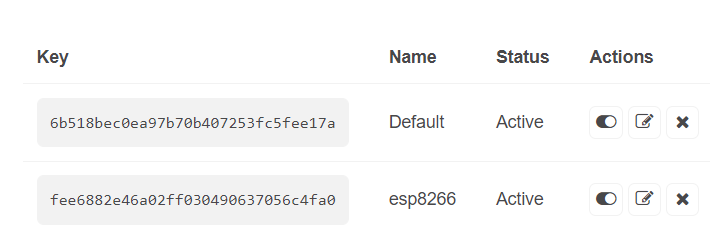
# Bài thực hành 3: Giám sát dữ liệu đám mây

1. Lấy dữ liệu trực tuyến (dữ liệu bất kỳ) hiển thị qua LCD i2c
2. **Linh kiện**

* Esp8266
* Dây usb
* Màn hình LCD I2C 1602
* 4 dây cái cái

1. **Thực hành**

* Lấy dữ liệu từ [OpenWeatherMAP](https://home.openweathermap.org/api_keys)



* Vị trí Khoa Kỹ thuật Công nghệ:

<https://www.google.com/maps/@16.4586397,107.5809145,20.43z>

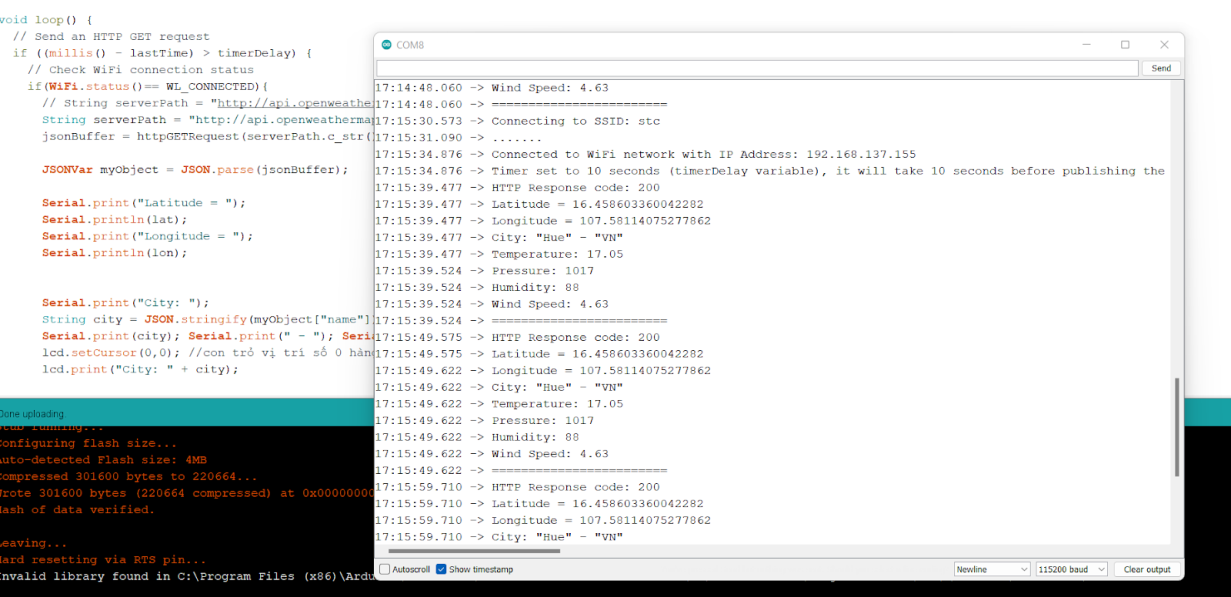
* Cú pháp:

http://api.openweathermap.org/data/2.5/weather?lat=xxx&lon=yyyy&units=metric&appid=api\_cua\_ban

* xxx, yyy = 16.458603360042282,107.58114075277862
* api\_cua\_ban = fee6882e46a02ff030490637056c4fa0

<http://api.openweathermap.org/data/2.5/weather?lat=16.458603360042282&lon=107.58114075277862&units=metric&appid=fee6882e46a02ff030490637056c4fa0>

* Màn hình Serial Monitor:



* Trình bày code:

#include <ESP8266WiFi.h>

#include <ESP8266HTTPClient.h>

#include <WiFiClient.h>

#include <Arduino\_JSON.h>

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,19,20);

const char\* ssid = "stc";    // tên wifi

const char\* password = "suthanhcon";  // mật khẩu wifi

// Your Domain name with URL path or IP address with path

String openWeatherMapApiKey = "fee6882e46a02ff030490637056c4fa0";

// Kinh độ, vĩ độ:

String lat = "16.458603360042282";

String lon = "107.58114075277862";

// THE DEFAULT TIMER IS SET TO 10 SECONDS FOR TESTING PURPOSES

// For a final application, check the API call limits per hour/minute to avoid getting blocked/banned

unsigned long lastTime = 0;

// Set timer to 10 seconds (10000)

unsigned long timerDelay = 10000;

String jsonBuffer;

void setup() {

  Serial.begin(115200);

  lcd.init(); //Khởi động LCD

  lcd.backlight(); //Mở đèn

  WiFiConnect();

  Serial.println("Timer set to 10 seconds (timerDelay variable), it will take 10 seconds before publishing the first reading.");

}

void loop() {

  // Send an HTTP GET request

  if ((millis() - lastTime) > timerDelay) {

    // Check WiFi connection status

    if(WiFi.status()== WL\_CONNECTED){

      // String serverPath = "http://api.openweathermap.org/data/2.5/weather?q=" + city + "," + countryCode + "&APPID=" + openWeatherMapApiKey;

      String serverPath = "http://api.openweathermap.org/data/2.5/weather?lat="+lat+"&lon="+lon+"&units=metric&appid=" + openWeatherMapApiKey;

      jsonBuffer = httpGETRequest(serverPath.c\_str());

      JSONVar myObject = JSON.parse(jsonBuffer);

      Serial.print("Latitude = ");

      Serial.println(lat);

      Serial.print("Longitude = ");

      Serial.println(lon);

      Serial.print("City: ");

      String city = JSON.stringify(myObject["name"]);

      Serial.print(city); Serial.print(" - "); Serial.println(myObject["sys"]["country"]);

      lcd.setCursor(0,0); //con trỏ vị trí số 0 hàng 1, hiện ô số 1

      lcd.print("City: " + city);

      Serial.print("Temperature: ");

      Serial.println(myObject["main"]["temp"]);

      lcd.setCursor(0,1); //con trỏ vị trí số 0 hàng 2, hiện ô số 1

      lcd.print("Temp: ");

      lcd.print(myObject["main"]["temp"]);

      lcd.print("C");

      Serial.print("Pressure: ");

      Serial.println(myObject["main"]["pressure"]);

      Serial.print("Humidity: ");

      Serial.println(myObject["main"]["humidity"]);

      Serial.print("Wind Speed: ");

      Serial.println(myObject["wind"]["speed"]);

      Serial.println("========================");

    }

    else {

      Serial.println("WiFi Disconnected");

    }

    lastTime = millis();

  }

}

void WiFiConnect(){

  WiFi.begin(ssid, password);

  Serial.print("Connecting to SSID: ");

  Serial.println(ssid);

  while(WiFi.status() != WL\_CONNECTED) {

    delay(500);

    Serial.print(".");

  }

  Serial.println("");

  Serial.print("Connected to WiFi network with IP Address: ");

  Serial.println(WiFi.localIP());

}

String httpGETRequest(const char\* serverName) {

  WiFiClient client;

  HTTPClient http;

  // Your IP address with path or Domain name with URL path

  http.begin(client, serverName);

  // Send HTTP POST request

  int httpResponseCode = http.GET();

  String payload = "{}";

  Serial.print("HTTP Response code: ");

  Serial.println(httpResponseCode);

  payload = http.getString();

  // Free resources

  http.end();

  return payload;

}

* Kết quả:

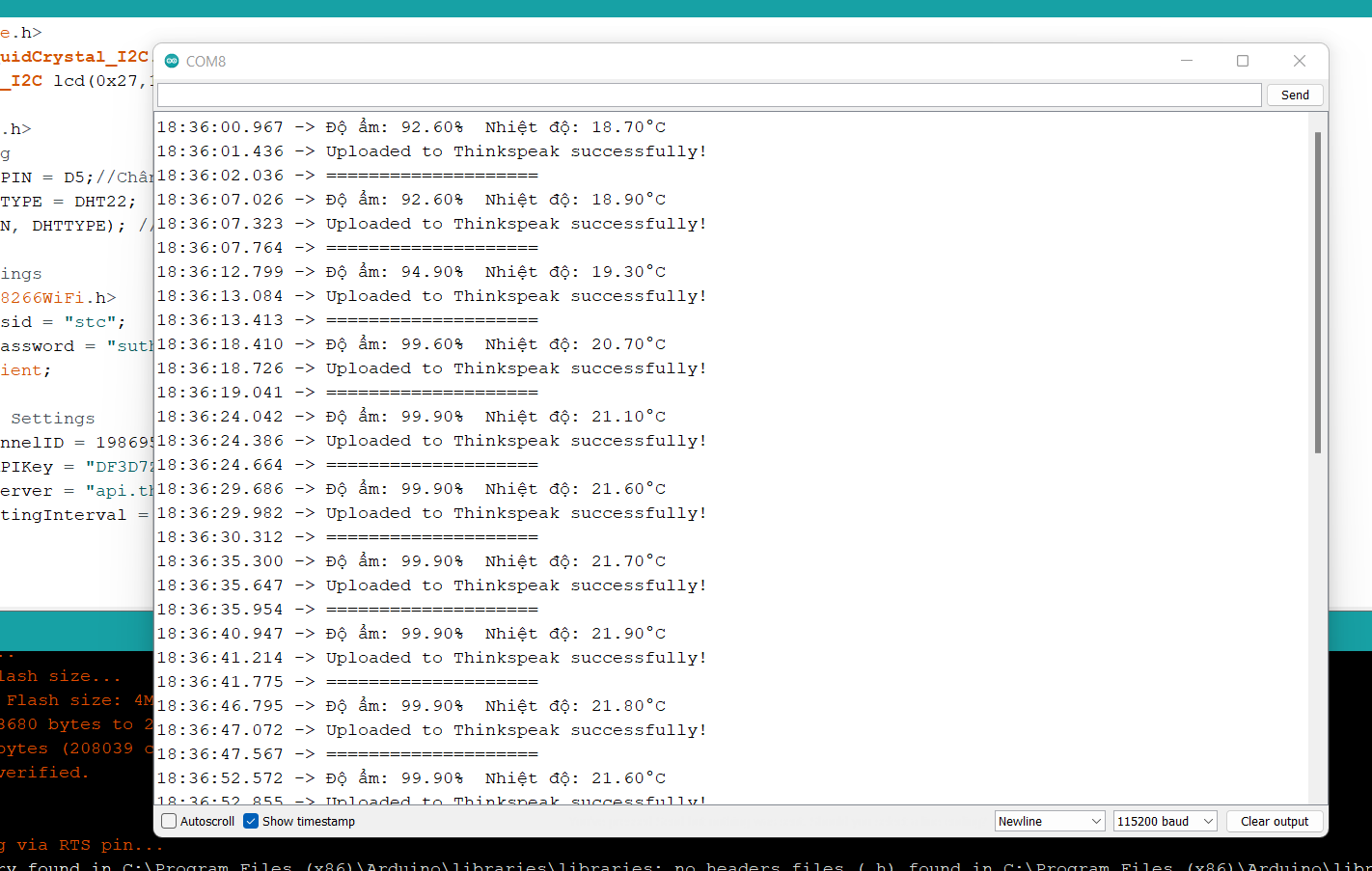


1. Kết nối cảm biến (DHT11/DHT22) với ESP8266 và hiển thị qua LCD i2c và lưu trữ dữ liệu trên Thingspeak (4)
2. **Linh kiện**

* Esp8266
* Dây usb
* Màn hình LCD I2C 1602
* 7 dây cái-cái
* DHT22

1. **Thực hành**

* Màn hình Serial Monitor:



* Trình bày code:

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,19,20); ///LCD config

#include <DHT.h>

/// DHT config

const int DHTPIN = D5;//Chân Out của cảm biến nối chân số 5 ESP8266

const int DHTTYPE = DHT22;   // Khai báo kiểu cảm biến là DHT21

DHT dht(DHTPIN, DHTTYPE); //Khai báo thư viện chân cảm biến và kiểu cảm biến

/// WiFi Settings

#include <ESP8266WiFi.h>

const char\* ssid = "stc";    // tên wifi

const char\* password = "suthanhcon";  // mật khẩu wifi

WiFiClient client;

// ThingSpeak Settings

const int channelID = 1986959;

String writeAPIKey = "DF3D7Z3WY634UCAA"; // write API key for your ThingSpeak Channel

const char\* server = "api.thingspeak.com";

const int postingInterval = 5 \* 1000; // post data every 5 seconds

void setup()

{

  Serial.begin(115200);

  dht.begin(); //Khởi động cảm biến

  WiFiConnect(); // Kết nối với WiFi

  lcd.init(); //Khởi động LCD

  lcd.backlight(); //Mở đèn

  lcd.setCursor(0,0); //Con trở ở vị trí 0 hàng 1, hiện ô 1

  lcd.print("DO AM:");

  lcd.setCursor(0,1);

  lcd.print("NHIET DO:");

}

void loop()

{

  // wait and then post again

  delay(postingInterval);

  float humi = dht.readHumidity(); //Đọc độ ẩm

  float temp = dht.readTemperature(); //Đọc nhiệt độ C

  // Kiểm tra cảm biến có hoạt động hay không

  if (isnan(humi) || isnan(temp))

  {

    Serial.println("Không có giá trị trả về từ cảm biến DHT");

    return;

  }

  Serial.print("Độ ẩm: ");

  Serial.print(humi);

  lcd.setCursor(7,0); //con trỏ vị trí số 7 hàng 1, hiện ô số 8

  lcd.print(humi);

  lcd.setCursor(12,0); //Con trở ở vị trí 12, hiện ô 13

  lcd.print("%");

  Serial.print("%  Nhiệt độ: ");

  Serial.print(temp);

  Serial.println("°C");

  lcd.setCursor(9,1);  //Con trở ở vị trí 9 hàng 2, hiện ô 10

  lcd.print(temp);

  lcd.print("'C");

  if (client.connect(server, 80)) {

    // Construct API request body

    String body = "field1=" + String(temp, 1) + "&field2=" + String(humi, 1);

    client.print("POST /update HTTP/1.1\n");

    client.print("Host: api.thingspeak.com\n");

    client.print("Connection: close\n");

    client.print("X-THINGSPEAKAPIKEY: " + writeAPIKey + "\n");

    client.print("Content-Type: application/x-www-form-urlencoded\n");

    client.print("Content-Length: ");

    client.print(body.length());

    client.print("\n\n");

    client.print(body);

    client.print("\n\n");

    Serial.println("Uploaded to Thinkspeak successfully!");

  }

  client.stop();

  Serial.println("====================");

}

void WiFiConnect(){

  WiFi.begin(ssid, password);

  Serial.print("Connecting to SSID: ");

  Serial.println(ssid);

  while(WiFi.status() != WL\_CONNECTED) {

    delay(500);

    Serial.print(".");

  }

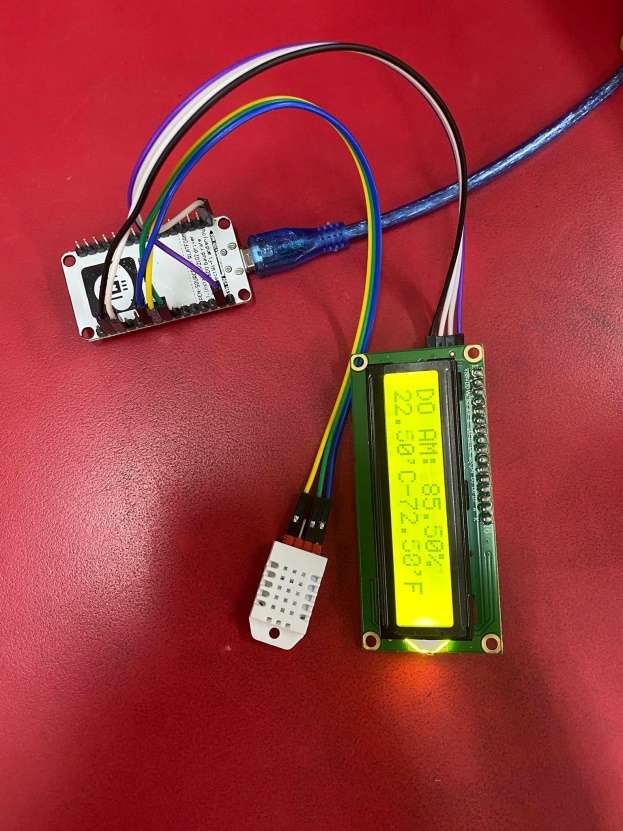
  Serial.println("");

  Serial.print("Connected to WiFi network with IP Address: ");

  Serial.println(WiFi.localIP());

}

* Kết quả:

  *\* (Mạch nối giống 2.3)*

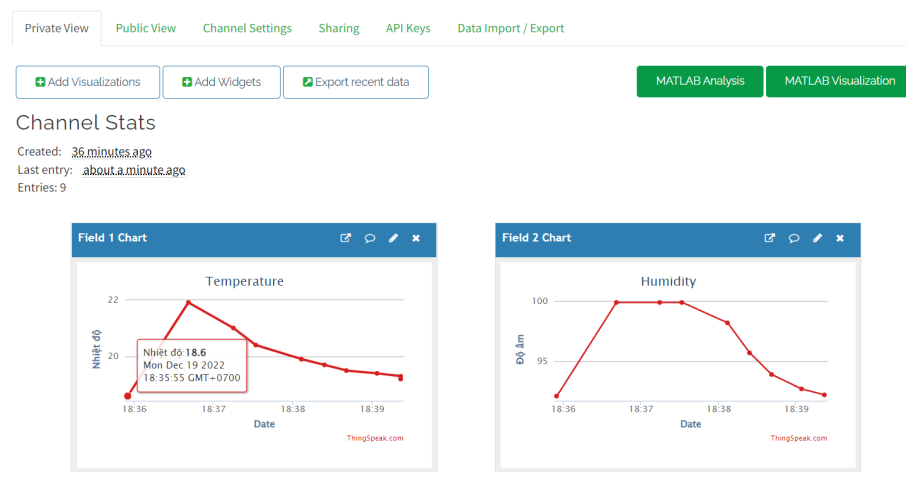
# Bài thực hành 4: Tương tác với các dịch vụ web và máy với máy

1. Giám sát dữ liệu cảm biến từ bảng điều khiển đám mây(lấy kết quả từ 3.2.)
2. **Linh kiện**

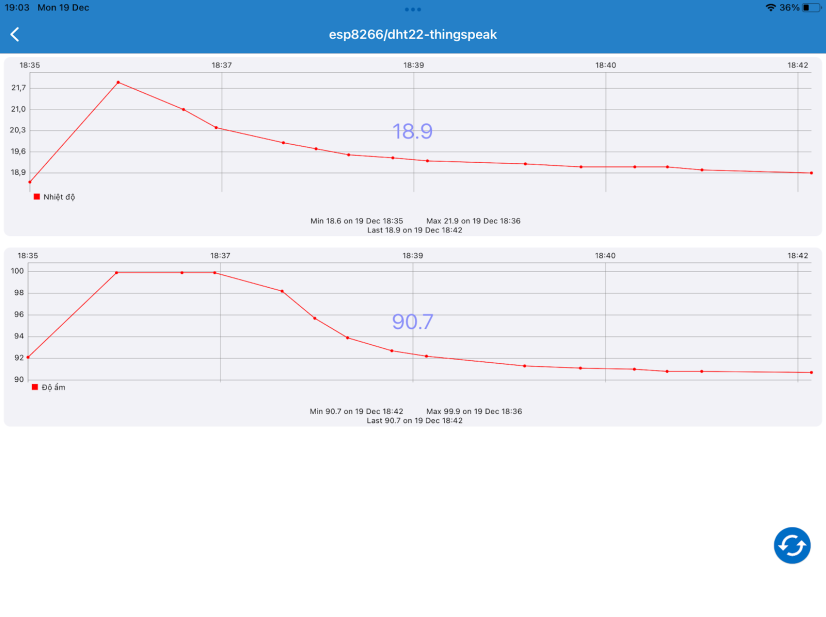
* Esp8266
* Dây usb
* Màn hình LCD I2C 1602
* 7 dây cái-cái
* DHT22

1. **Thực hành**

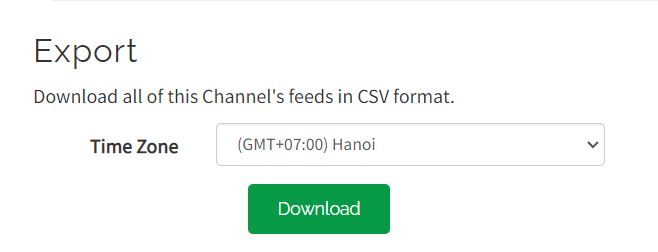
* Giao diện trên ThingSpeak

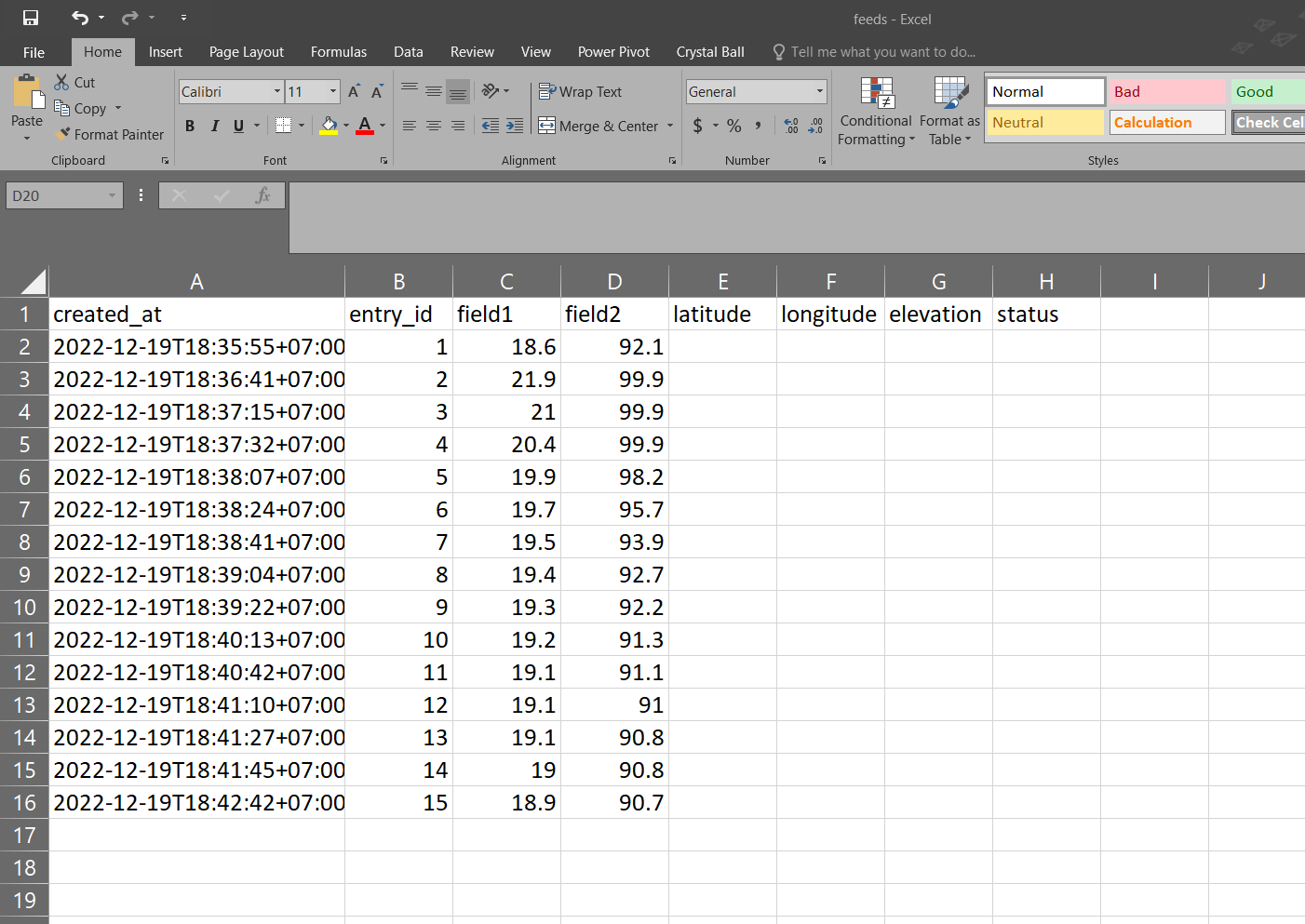


* Giao diện trên App ThinkView

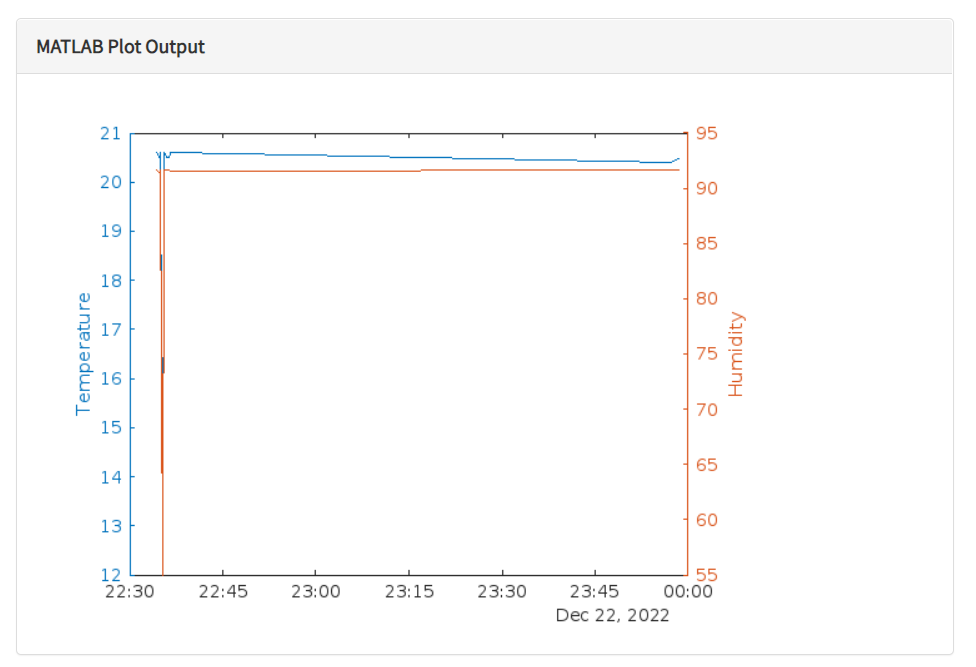


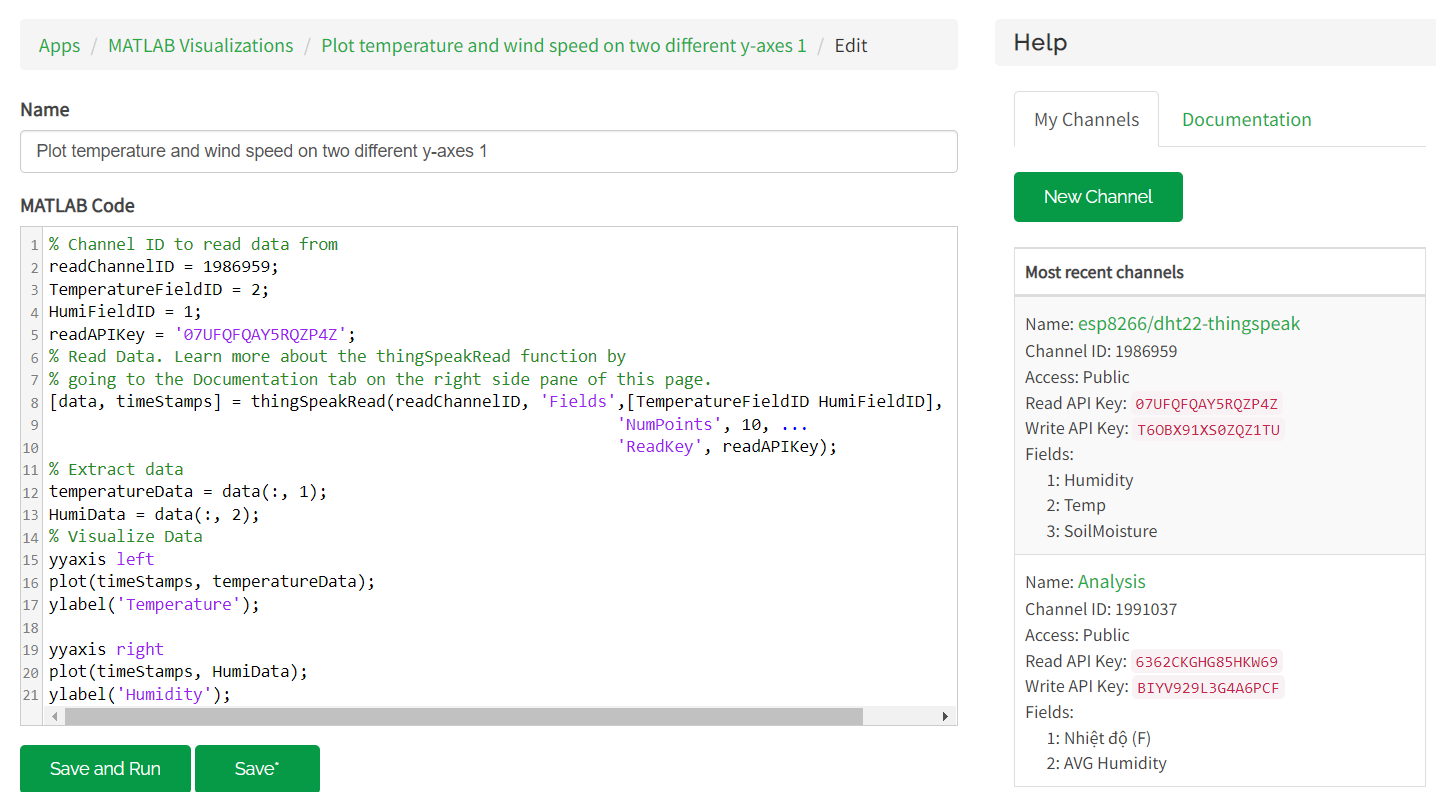
* Xuất dữ liệu từ ThinkSpeak và lưu thành file.CSV

****

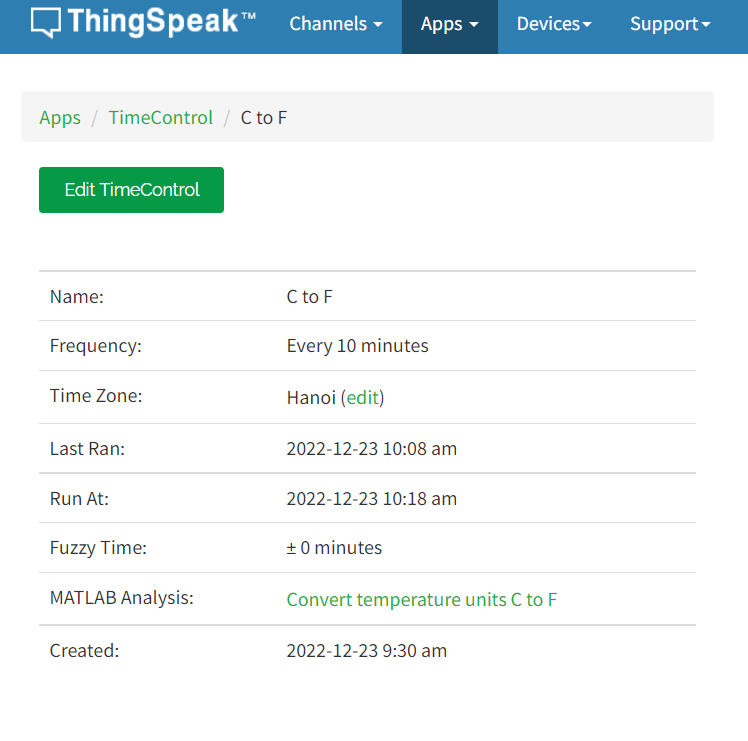


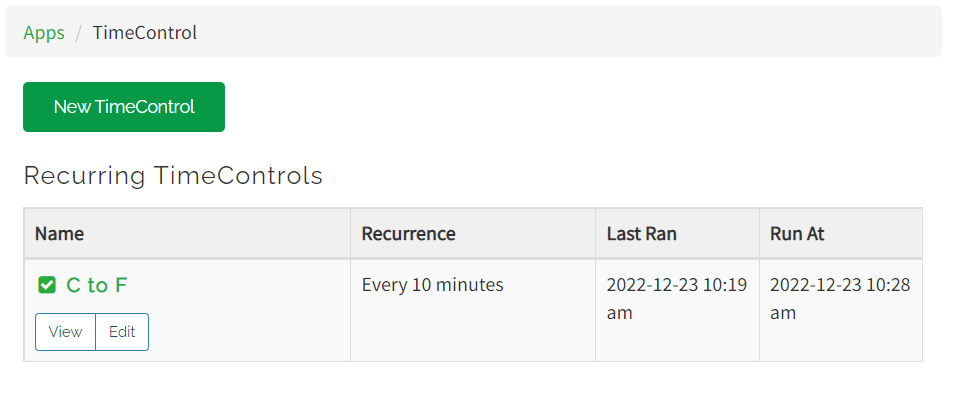
* MATLAB Visualizations



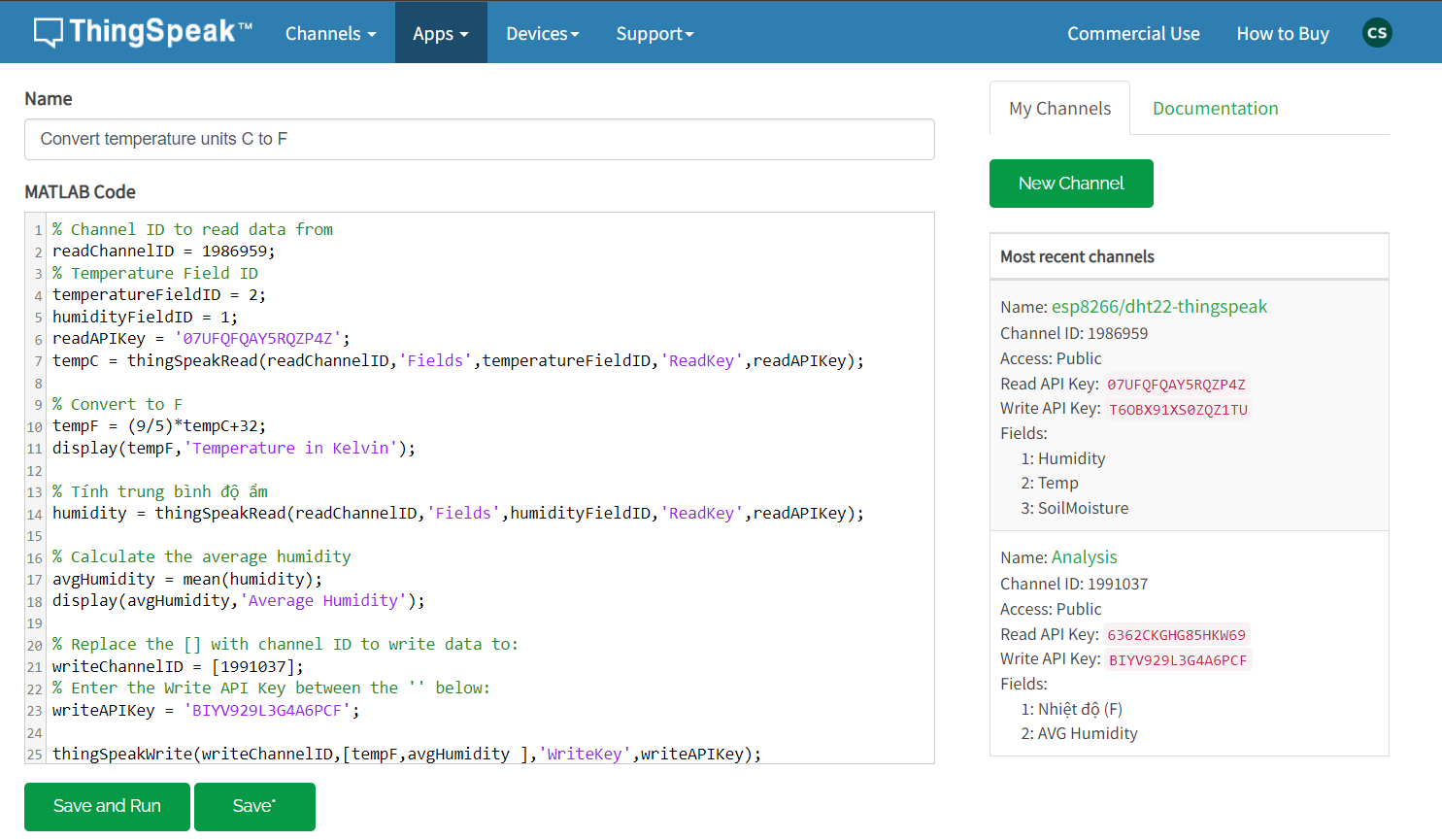


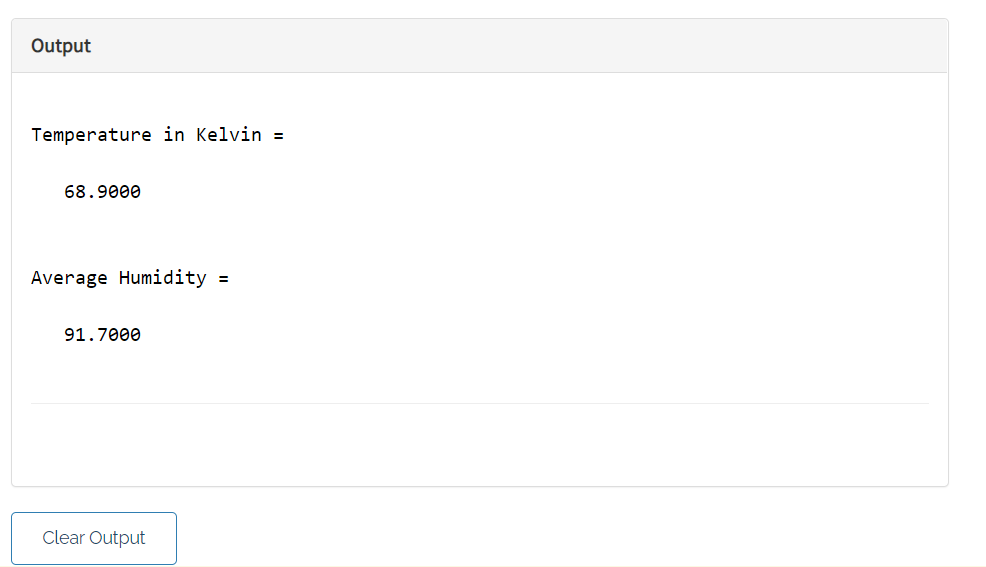
* TimeControl





* MATLAB Analysis





# Tài liệu tham khảo

1. chớp tắt đèn LED. *arduino.esp8266.vn.* [Online] [Cited: 12 20, 2022.] https://arduino.esp8266.vn/basic/led.html.

2. ESP8266 ADC – Read Analog Values with Arduino IDE, MicroPython and Lua. *randomnerdtutorials.com.* [Online] [Cited: 12 20, 2022.] https://randomnerdtutorials.com/esp8266-adc-reading-analog-values-with-nodemcu/.

3. ESP8266 Weather Server With LCD Display. *www.instructables.com.* [Online] [Cited: 12 20, 2022.] https://www.instructables.com/ESP8266-Weather-Server-With-LCD-Display/.

4. Đo nhiệt độ, độ ẩm và gởi lên Thingspeak. *arduino.esp8266.vn.* [Online] 12 20, 2022. https://arduino.esp8266.vn/projects/dht11-thingspeak.html.